Scientometrics manuscript No.
(will be inserted by the editor)

Exploratory Search of Academic Publication and Citation Data using Interactive Tag Cloud Visualizations

Marcel Dunaiski · Gillian J. Greene · Bernd Fischer

The final publication is available at Springer via http://dx.doi.org/10.1007/s11192-016-2236-3

Abstract Acquiring an overview of an unfamiliar discipline and exploring relevant papers and journals is often a laborious task for researchers.

In this paper we show how exploratory search can be supported on a large collection of academic papers to allow users to answer complex scientometric questions which traditional retrieval approaches do not support optimally. We use our ConceptCloud browser, which makes use of a combination of concept lattices and tag clouds, to visually present academic publication data (specifically, the ACM Digital Library) in a browsable format that facilitates exploratory search. We augment this dataset with semantic categories, obtained through automatic keyphrase extraction from papers' titles and abstracts, in order to provide the user with a uniform keyphrases of the underlying data collection. We use the citations and references of papers to provide additional mechanisms for exploring relevant research by presenting aggregated reference and citation data not only for a single paper but also across topics, authors and journals, which is novel in our approach.

We conduct a user study to evaluate our approach in which we asked 34 participants, from different academic backgrounds with varying degrees of research experience, to answer a variety of scientometric questions using our ConceptCloud browser. Participants were able to answer complex scientometric questions using our ConceptCloud browser with a mean correctness of 73%, with the user's prior research experience having no statistically significant effect on the results.

Marcel Dunaiski
Computer Science Division, Stellenbosch University, Matieland, South Africa
E-mail: marcel@ml.sun.ac.za

Gillian J. Greene
CAIR, CSIR Meraka
Computer Science Division, Stellenbosch University, Matieland, South Africa
E-mail: ggreene@cs.sun.ac.za

Bernd Fischer
CAIR, CSIR Meraka
Computer Science Division, Stellenbosch University, Matieland, South Africa
E-mail: bfischer@cs.sun.ac.za

Keywords Tag Clouds · Exploratory Search · Formal Concept Analysis · Automatic Keyphrase Extraction · Citation Analysis · User Study

1 Introduction

The large amount of academic papers that is available for an increasingly large number of topics makes it difficult for researchers to keep up to date with a field or to find relevant related work (Parolo et al. 2015). Traditional approaches for finding relevant academic publications are search-based and rely on users to manually specify keywords of interest. Other mechanisms for researchers to discover papers of interest (for example, Mendeley's (Hoey 2015) or Google Scholar's (Connor, James 2012) paper recommendations) are also becoming increasingly relevant. However, while automatic paper suggestions can be used to highlight possibly relevant work for researches, they make use of a researcher's academic profile and therefore cannot support novice researchers, or those unfamiliar with a topic, in finding the most relevant work on new topics.

While publication data can be made available to users in different formats (e.g., paper suggestions or a searchable paper collection), all tasks are typically supported by an indexed collection of papers which has been classified. Keyphrases are often assigned to papers for categorization when they are submitted for publication, and papers can even be manually classified into topics such as ACM's classification scheme (Accociation for Computing Machinery 2015). However, author-assigned keywords are not always available and often not comprehensively descriptive enough for search tasks (Zhang et al. 2016). Moreover, author-assigned keyphrases are often not uniform for all papers of the same topic which makes retrieving papers using author-assigned keyphrases impractical, as multiple searches, specified slightly differently, may be required to obtain a more complete overview for a single topic.

Search-based approaches for interacting with publication data are widely supported, but when a researcher is unfamiliar with a research field or has not yet formulated a direct query, their task becomes one of exploratory search (Marchionini 2006; White and Roth 2009) rather than direct retrieval. Exploratory search approaches are concerned with maximizing recall (i.e., showing all relevant information) as opposed to maximizing precision (i.e., not showing any irrelevant information) which is the main concern of retrieval approaches (Marchionini 2006).

In this paper we present and evaluate an exploratory search tool for a large academic publication dataset, the ACM Digital Library (2016). To facilitate exploration of the data we need a uniform categorization of the papers which is not provided by the author-assigned keywords or keyphrases. We therefore use an unsupervised approach to extract relevant keyphrases directly from the titles and abstracts of academic papers. We choose an unsupervised approach because test data that is more comprehensive than the manually assigned author keywords, would need to be manually collected for each paper which renders supervised approaches impractical for the keyphrase coverage required.

To support exploration of an academic publication dataset we construct a concept lattice from the papers' meta-data (e.g., authors, affiliations, publication years and publishing venues) and the extracted keyphrases as concept lattices have been shown to be useful for browsing and exploration tasks (Fischer 2000; Lindig

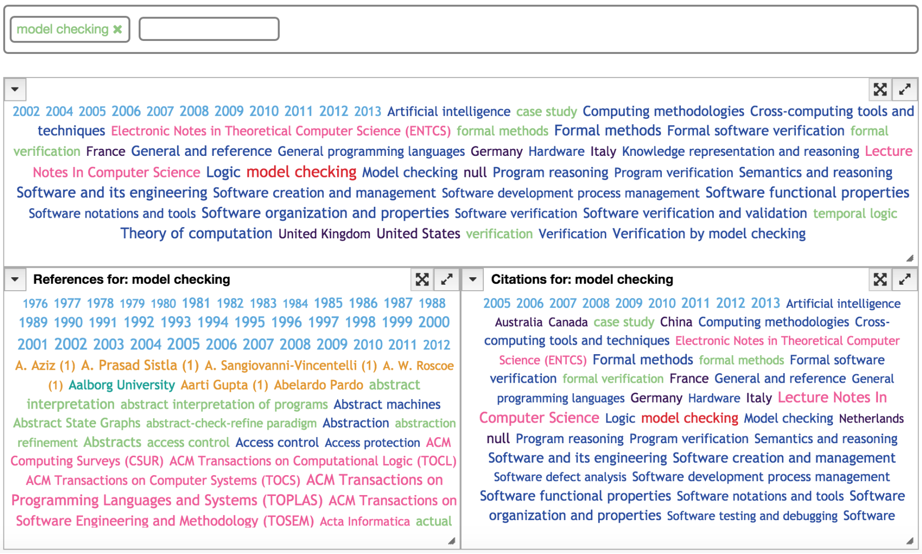


Fig. 1: Three tag clouds with a selected tag (red), authors (orange), years (cyan), and keyphrases (green). For each tag cloud the 50 largest tags are shown from papers associated with the keyphrase “Model Checking” (top), papers that are referenced by papers associated with the keyphrase “Model Checking” (bottom left), and papers that cite papers associated with the keyphrase “Model Checking”.

1995; Carpineto and Romano 1996). However, since diagrams of large concept lattices are difficult to read and interpret, we construct a more scalable, interactive tag cloud on top of the concept lattice to provide the user with a more intuitive interface to the underlying data. Figure 1 shows a tag cloud view of papers with the key-phrase “model checking”. Tag clouds are a simple textual visualization where the frequency of a word is indicated by its size in the tag cloud. We use tag clouds to present the aggregated information from the underlying meta-data of the academic papers. Our tag clouds are interactive, i.e., the user can refine the view of the publication dataset by selecting appropriate combinations of tags. The view is then updated to only show the aggregated information of the publications to which the selected tags apply. This facilitates the exploration of the data since the user is always presented with aggregated information about publications associated with current tag selections which in turn can be used to make further refinements.

A list of publications associated with the current tag selections is also available to the user, so that they can use the paper’s title, author names or journal to retrieve the paper’s full-text. However, unlike traditional list-based interfaces, we allow the user to make continued refinements of the data until the list of relevant publications is small enough to make examination of the full-texts feasible, instead of presenting large lists of publications that all need to be manually examined.

In addition to relevant keyphrases, a paper’s references and citations are also a valuable source of information when trying to find related research. In traditional digital library formats, references and citations are currently only provided in a list view for a single paper at a time. In our approach we use citation and reference data to provide a powerful mechanism for exploring relevant research. We allow users to view aggregated reference and citation information for a single paper, for example, the user can quickly get a summary of what topics of papers have referenced or cited a specific paper. We also allow users to view aggregated citation and reference data not only for a single paper at a time but for all papers that have a selected property. For example users can view aggregated citation or reference information of all papers by a certain author, all papers in a particular conference or all papers on a certain topic. Figure 1 shows an example of our tag cloud browser which shows the tags associated to the selected key-phrase (model checking) as well as the tag clouds of papers that have been referenced by "model checking" papers (bottom left) and papers that have cited "model checking" papers (bottom right).

In this paper we present our approach for supporting exploratory search of academic paper collections using a tag cloud visualization with an underlying concept lattice to support navigation. We show that our approach provides an alternative to traditional retrieval and indexing approaches for academic publication data and allowed users in a study to answer complex scientometric questions with an mean correctness percentage of 73%. Through our user study and analyzing the participants' efficiency in answering questions, we see that there is a learning effect for new users of the browser when they are slower than on average to answer the provided questions. However, after a short learning period (approximately 18 minutes), participants were able to effectively use our ConceptCloud browser.

In this paper we make the following contributions:

- We use a novel combination of concept lattices with tag clouds to index and visualize a large academic publication dataset.
- We demonstrate an approach for visualizing aggregated information from citations and references.
- We combine previous approaches to provide a key-phrase extraction technique to extract uniform key-phrases for academic publication collections
- We showcase the feasibility of using an exploratory search tool to answer complex scientometric questions.
- We demonstrate the visible learning effects of an exploratory search tool through a controlled user study.

In this paper we first describe our key-phrase extraction approach which combines a number of previous approaches (see Section 2). We then describe our approach of constructing interactive tag clouds for the academic publication data (see Section 3). We conduct a controlled user study to evaluate whether untrained users are able to use our browser to answer questions about a digital library dataset (see Section 4). We show that, while there is a noticeable learning effect, users are able to effectively answer complex scientometric questions using our approach with a mean accuracy of 73% (Section 5 and 6). In Section 7 we discuss other publication browsing techniques in relation to our approach.

2 Keyphrase Extraction for Academic Papers

The usual motivation to extract or assign keywords and keyphrases to documents is to classify them into topics, create summaries, or index the documents to support search tasks. A good set of keywords or keyphrases describes a document well (good coverage), is distinctive enough not to be applicable to too many documents (relevant), and easily understandable to people (descriptive).

In order to acquire keywords or keyphrases for a certain document, the following two steps are typically involved. Firstly, candidate words or phrases are extracted from the document using a combination of heuristic rules such as using a stop word list to remove stop words (Liu et al. 2009), using words with certain part-of-speech tags (Mihalcea and Tarau 2004; Wan and Xiao 2008), identifying n-grams (Witten et al. 1999; Hulth 2003; Medelyan et al. 2009) or noun phrases that follow pre-defined lexico-syntactic patterns (Hulth 2003; Nguyen and Kan 2007), or matching word sequences to external knowledge bases such as Wikipedia (Grineva et al. 2009; Liu et al. 2009; Medelyan et al. 2009) or WordNet (Li et al. 2006). The main problem of these approaches is that the list of candidates can become very long, especially with long articles where full-texts are available. If however, white-lists such as Wikipedia article titles are used, the candidate list becomes restricted.

The second step involves classifying the candidate words and phrases as keywords and keyphrases. This can be achieved through classification (Witten et al. 1999; Frank et al. 1999; Hulth 2003; Medelyan et al. 2009), ranking of phrases where the top ranked phrases are selected as keyphrases (Mihalcea and Tarau 2004; Wan and Xiao 2008; Jiang et al. 2009), or through pruning of phrases that are unlikely to be keyphrases (Huang et al. 2006; You et al. 2009).

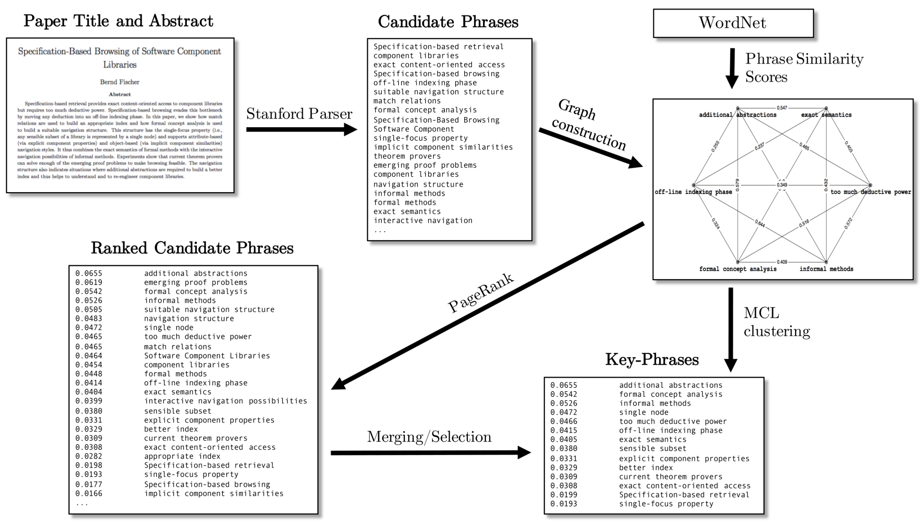


Fig. 2: An overview of our keyphrase extraction procedure which combines a number of previously proposed techniques.

In this paper we use an unsupervised approach for keyphrase extraction that combines a number of previously proposed techniques (Miller 1995; Li et al. 2006; Van Dogen 2000; Brin and Page 1998). An overview of our keyphrase extraction procedure is given in Figure 2. Candidate keyphrases are noun-phrases that are extracted from paper titles and abstracts and filtered based on a set of heuristic rules. The keyphrase selection step applies ranking and clustering based on semantic similarity between candidate keyphrases. The following sections describe this approach in more detail.

2.1 Candidate Word and Phrase Identification

In order to identify candidate words and phrases, the Stanford Parser (Klein and Manning 2003) is used to parse each sentence of a document and extract noun-phrases that are shorter than six words. Furthermore, heuristic rules are applied to filter out improbable keyphrases. For example, if an identified noun-phrase consists of a single word, it is only included into the candidate phrase set if it is a proper singular noun. Moreover, certain words within candidate phrases are excluded such as determiners, personal and possessive pronouns, and numerals.

A graph is constructed from the candidate phrases in which the edge connecting two nodes is weighted according to the similarity between the phrases associated with the two nodes. We use WordNet (Miller 1995) as a basis for the similarity computation between phrases (Li et al. 2006), Markov Chain Clustering (MCL) (Van Dogen 2000) to cluster similar keyphrases, and PageRank (Brin and Page 1998) to rank the phrases according to their importance.

2.2 Word Similarity

WordNet is a lexical database of English words in which nouns, verbs, adjectives and adverbs are organized into sets of synonyms, where each group (synset) describes a distinct lexicalized concept (Miller 1995). The power of WordNet is that synsets are linked to indicate conceptual or semantic relations.

The semantic similarity between two words is computed by locating them in WordNet and using its hierarchical structure to compute their distance. A simple approach is to find the minimum path length connecting the two words. In larger semantic nets such as WordNet, however, this approach is not very accurate since it does not take the depth of the hierarchy at which the words were matched into consideration.

In Figure 3 one can see that the distance between “girl” and “animal” and the distance between “girl” and “teacher” are both 5, although “girl” is arguably more closely related to “teacher” than “animal”. This problem is addressed by taking the depth of the lowest common subsumer (i.e., most specific ancestor node) into consideration. In this example, the lowest common subsumer of both “girl” and “animal” is “organism” with a depth of 6, compared to the lowest common subsumer “adult” with a depth of 8 for the words “girl” and “teacher”.

As described by Li et al. (2006), the resulting function for calculating the similarity between two words is:

$$sim_{word}(w_i, w_j) = e^{-\alpha l} \cdot \frac{e^{\beta d} - e^{-\beta d}}{e^{\beta d} + e^{-\beta d}}$$

where l is the shortest distance between the two words w_i and w_j, and d is the depth from the root to their lowest common subsumer. The parameters α and

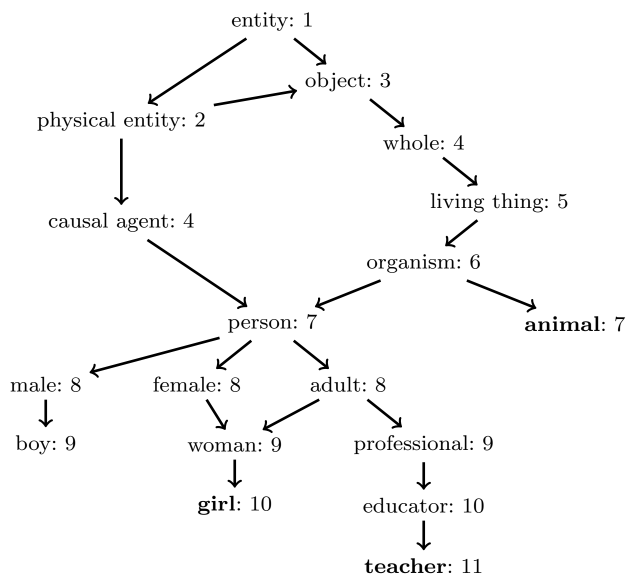


Fig. 3: WordNet hierarchy example. The root node is "entity" with an associated depth of 1 as indicated. The node "teacher" is at depth 11.

β control the weighting of the path length and the depth of the lowest common subsumer, respectively. These parameters can be tuned depending on the semantic network used. According to Li et al. (2003), the parameters' optimal values for WordNet are $\alpha = 0.2$ and $\beta = 0.45$.

2.3 Phrase Similarity

Given two sentences S_1 and S_2, let $\boldsymbol{s} = \{S_1 \cup S_2\}$ be the lexical vector containing the set of unique words in S_1 and S_2 and therefore contain the semantic context of the two sentences.

For each sentence S_1 and S_2 a semantic vector $\boldsymbol{s}_i$ is constructed from the combined semantic vector $\boldsymbol{s}$, which has the same length as $\boldsymbol{s}$. Considering sentence S_1, $\boldsymbol{s}_1$ is constructed by finding the most similar word in S_1 for each term in $\boldsymbol{s}$ and inserting their similarity score. Therefore, if the term appears in the sentence S_1, the corresponding value is set to 1. Otherwise, the similarity score of the most similar term in the joint list is stored in the vector, given that the similarity score is above the threshold $\phi = 0.2$, otherwise the value is set to 0 (Li et al. 2006).

Since different words in a phrase contribute toward its meaning in different degrees, each word t in the vectors $\boldsymbol{s}_i$ is also weighted by its importance to the paper d from which it was extracted. For this, the *tf-idf* value for the word t in document d is computed as follows:

$$tf\text{-}idf(t,d) = (1 + log(f_{t,d})) \cdot log\left(\frac{|D|}{1 + |\{d \in D : t \in d\}|}\right)$$

where $f_{t,d}$ is the frequency of word t in paper d and D is the set of papers. WordNet lemmatization and Porter stemming (Porter 1980) are applied to each word in the *tf-idf* corpus.

Considering sentence S_1 again, the value of an entry s_i in the semantic vector $\boldsymbol{s}_1$ associated with word w_i is weighted by multiplying it with the *tf-idf* of the word w_i in the sentence and the most similar word w_i^* in the semantic vector $\boldsymbol{s}$, i.e. $s_i = s_i \cdot \mathit{tf\text{-}idf}(w_i, d) \cdot \mathit{tf\text{-}idf}(w_i^*, d)$, where d is the current documents for which keyphrase similarity scores are computed.

Finally, the resulting similarity score between two phrases is the cosine distance between the two vectors $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$:

$$sim_{phrase}(S_1, S_2) = \frac{\boldsymbol{s}_1 \cdot \boldsymbol{s}_2}{\|\boldsymbol{s}_1\| \cdot \|\boldsymbol{s}_2\|}$$

The word order in a phrase is also important when comparing two phrases and therefore a word order similarity between two phrases is computed. Again let S be the set of all unique words in sentences S_1 and S_2 where each word is associated with a unique index. Let $\boldsymbol{r}_1$ be the word order vector for sentence S_1. Then for each word in S, find the most common word in S_1 and insert the corresponding index of the word into $\boldsymbol{r}_1$. The same is done for S_2 to obtain $\boldsymbol{r}_2$. The word order similarity score between the two phrases is then:

$$sim_{order}(S_1, S_2) = 1 - \frac{\|\boldsymbol{r}_1 - \boldsymbol{r}_2\|}{\|\boldsymbol{r}_1 + \boldsymbol{r}_2\|}$$

The final similarity score of the sentences S_1 and S_2 is a convex combination of the phrase semantic similarity and the phrase order similarity scores:

$$sim(S_1, S_2) = \delta \cdot \frac{\boldsymbol{s}_1 \cdot \boldsymbol{s}_2}{\|\boldsymbol{s}_1\| \cdot \|\boldsymbol{s}_2\|} + (1 - \delta) \cdot \left(1 - \frac{\|\boldsymbol{r}_1 - \boldsymbol{r}_2\|}{\|\boldsymbol{r}_1 + \boldsymbol{r}_2\|}\right)$$

where δ controls the weight associated with the two similarity scores and is set to 0.85 (Li et al. 2006).

2.4 Candidate Phrase Ranking

Let P be the set of all candidate phrases of a paper. Then a weighted undirected graph G can be constructed such that each edge $E(p_i, p_j) = sim(p_i, p_j)$ is the similarity between each two phrases p_i and p_j, given that their similarity score is larger than 0.

PageRank (Brin and Page 1998) is used to rank all the candidate phrases according to their importance. PageRank was initially developed to rank web pages according to their importance or relevance and uses the graph structure of the Internet for its computation. The idea is that links from important websites should weigh more than links from unimportant websites. The result of the PageRank computation is a probability distribution that represents the likelihood that a web surfer who is randomly clicking on links will arrive at a certain web page. It has been shown that PageRank works well in many applications where the goal is to identify important nodes in a graph (Brin and Page 1998; Chen et al. 2007; West et al. 2010, 2013; Gollapalli and Caragea 2014; Dunaiski et al. 2016).

At each iteration t of the algorithm the PageRank value PR for a phrase p_i is calculated using the following formula:

$$PR_t(p_i) = \frac{1-\alpha}{n} + \alpha \cdot \sum_{p_j \in N(p_i)} \frac{PR_{t_1}(p_j)}{\mathrm{d}(p_j)}$$

where n is the number of candidate phrases, $\mathrm{d}(p_j)$ is the degree of the node associated with phrase p_j and $N(p_i)$ is the neighbourhood of the node associated with the phrase p_i. The PageRank algorithm is initialized with $PR_0(p) = \frac{1}{n}$. The computation stops when a predefined threshold $\delta = 10^{-6}$ is reached:

$$\sum_p \|PR_t(p) - PR_{t-1}(p)\| < N \cdot \delta$$

2.5 Candidate Phrase Clustering

After the candidate phrases have been ranked, similar phrases are clustered together using the Markov Chain Cluster (MCL) algorithm (Van Dogen 2000). The MCL algorithm is based on random walks on the graph with the intuition that when a random walker starts at a node and follows an edge, she is more likely to stay within a cluster of nodes than follow an edge that leads her to another cluster.

With each iteration of the power method, strong links are strengthened and less popular neighbors are demoted. This process is amplified with each iteration by taking each column of the matrix to a non-negative power $r = 2.6$ in an *inflation* operation. The *inflation* operation is alternated with an *expansion* operation, in which the e-th power of the matrix is taken ($e = 2$). *Inflation* controls the granularity of the clusters, while *expansion* is responsible for allowing flow to connect different regions of the graph. After each iteration the matrix is re-normalized to keep it stochastic to ensure that the method converges.

The computation stops when the matrix is near-idempotent (Van Dogen 2000, p. 55), i.e., $max\left\{M^2 - M\right\} < \delta$, where δ is the above mentioned precision threshold.

The last step is to find the clusters from the new idempotent matrix M. This is done by looking at the positive flows of each node. If a node has at least one positive flow (a positive value in its corresponding row) it is called an attractor. All nodes that have a positive flow with an attractor are clustered together. For each cluster the node with the highest rank is chosen.

Using the method described in this section, each paper is processed individually by extracting candidate keyphrases from the title and the abstract. Semantic similarity values are computed for these candidate keyphrases which are used for ranking and clustering to obtain a list of keyphrases for an individual paper. We then lemmatize and lookup each word in WordNet, to obtain a more uniform distribution of keyphrases over all papers. We then assign the resulting keyphrases to each paper which can be used to support navigation in the ConceptCloud browser.

3 Navigating Concept Lattices With Tag Clouds

We use a concept lattice to index the publication dataset and support navigation in the dataset. Concept lattices provide a large amount of internal structure for the data and have been shown to be useful for exploration or browsing tasks (Fischer 2000; Lindig 1995; Carpineto and Romano 1996). However, since diagrams of large concept lattices are difficult to visualize and interpret they do not provide a suitable interface with which to navigate in the underlying dataset. Therefore, we provide a more intuitive tag cloud interface in which navigation is facilitated by the underlying concept lattice. Selections (and de-selections) in the tag cloud drive navigation operations in the underlying concept lattice. Figure 4 shows an overview of our approach.

3.1 Formal Concept Analysis

Formal concept analysis (FCA) (Wille 1982; Ganter and Wille 1999; Davey and Priestley 2002) uses lattice-theoretic methods to investigate abstract relations between objects and their attributes. Such *contexts* can be imagined as cross tables where the rows are objects and the columns are attributes.

Definition 1 *A formal* context *is a triple* $(\mathcal{O}, \mathcal{A}, \mathcal{I})$ *where* $\mathcal{O}$ *and* $\mathcal{A}$ *are sets of objects and attributes, respectively, and* $\mathcal{I} \subseteq \mathcal{O} \times \mathcal{A}$ *is an arbitrary incidence relation.*

Definition 2 *Let* $(\mathcal{O}, \mathcal{A}, \mathcal{I})$ *be a context,* $O \subseteq \mathcal{O}$*, and* $A \subseteq \mathcal{A}$*. The* common attributes *of* O *are defined by* $\alpha(O) = \{a \in \mathcal{A} \mid \forall o \in O : (o, a) \in \mathcal{I}\}$*, the* common objects *of* A *by* $\omega(A) = \{o \in \mathcal{O} \mid \forall a \in A : (o, a) \in \mathcal{I}\}$*.*

Concepts are pairs of objects and attributes which are synonymous. They are maximal rectangles (modulo permutation of rows and columns) in the context table. For our purpose, academic papers form the objects in the context, while the attributes in the context table consist of information associated with each paper, such as the authors, extracted keyphrases or the publication year.

Definition 3 *Let* $\mathcal{C}$ *be a context.* $c = (O, A)$ *is called a* concept *of* $\mathcal{C}$ *iff* $\alpha(O) = A$ *and* $\omega(A) = O$*.* $\pi_O(c) = O$ *and* $\pi_A(c) = A$ *are called* c*'s* extent *and* intent*, respectively. The set of all concepts of* $\mathcal{C}$ *is denoted by* $B(\mathcal{C})$*.*

Concepts are partially ordered by inclusion of extents such that a concept's extent includes the extent of all of its subconcepts; the intent-part follows by duality. Each concept consists of a set of academic papers (extent) and a set of attributes (intent) that are common to all papers in the extent. For example, we may observe a concept where the year 2016 forms the intent (attributes) and all papers published in 2016 will then be in the concept's extent.

Definition 4 *Let* $\mathcal{C}$ *be a context,* $c_1 = (O_1, A_1), c_2 = (O_2, A_2) \in B(\mathcal{C})$*.* c_1 *and* c_2 *are ordered by the* subconcept relation*,* $c_1 \leq c_2$*, iff* $O_1 \subseteq O_2$*. The structure of* $B(\mathcal{C})$ *and* $\leq$ *is denoted by* $\mathcal{B}(\mathcal{C})$*.*

In our approach, where academic papers themselves comprise the extent of each concept, the concept lattice stores the full set of papers in the extent of its top concept. Therefore, the top concept's intent contains only attributes that are common to all papers in the dataset and is therefore likely to be empty. Towards the bottom of the lattice the concepts have smaller extents (i.e., fewer academic papers) and larger intents (i.e., more attributes common to the papers in the concept's extent). At the atomic level of the concept lattice, a concept's intent contains all attributes associated with a single paper that forms the concept's extent.

The basic theorem of FCA states that the structure induced by the concepts of a formal context and their ordering is always a complete lattice. Such *concept lattices* have strong mathematical properties and reveal hidden structural and hierarchical properties of the original relation. They can be computed automatically from any given relation between objects and attributes. The greatest lower bound or *meet* and least upper bound or *join* can also be expressed by the common attributes and objects.

Theorem 5 (Wille (1982)) *Let $\mathcal{C}$ be a context. Then $\mathcal{B}(\mathcal{C})$ is a complete lattice, the* concept lattice *of $\mathcal{C}$. Its meet and join operation for any set $I \subset B(\mathcal{C})$ of concepts are given by*

$$\bigwedge_{i \in I} (O_i, A_i) = \left(\bigcap_{i \in I} O_i,\ \alpha(\omega(\bigcup_{i \in I} A_i)) \right)$$

$$\bigvee_{i \in I} (O_i, A_i) = \left(\omega(\alpha(\bigcup_{i \in I} O_i)),\ \bigcap_{i \in I} A_i \right)$$

Each attribute and object has a uniquely determined defining concept in the lattice. The defining concepts can be calculated directly from the attribute or object, respectively, and need not be searched in the lattice. If there are multiple papers which share all of their attributes then there will be more than one paper in the defining concept's extent. This concept will be the defining concept for all of the papers in the extent.

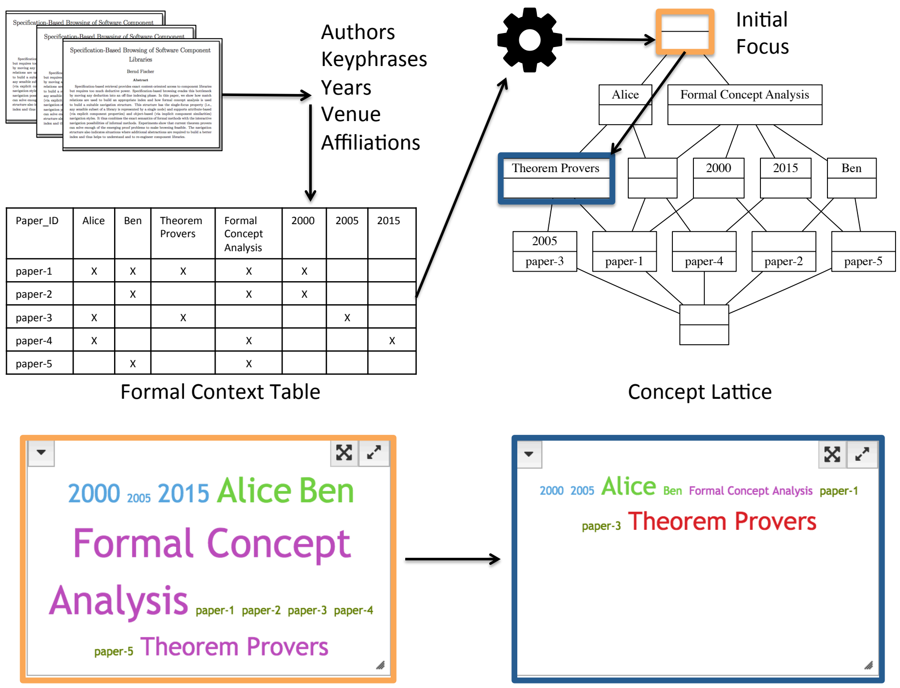
Definition 6 *Let $\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{I})$ be a concept lattice. The* defining concept *of an attribute $a \in \mathcal{A}$ (object $o \in \mathcal{O}$) is the greatest (smallest) concept c such that $a \in \pi_A(c)$ ($o \in \pi_O(c)$) holds. It is denoted by $\mu(a)$ ($\sigma(o)$). We use the $\delta(x)$ to denote $\mu(x)$ if x is an attribute and $\sigma(x)$ otherwise.*

3.1.1 Formal Context Construction

Computing formal contexts is an intermediate step in the construction of concept lattices. It should be noted that formal contexts can be constructed in various ways, depending on the choice of objects and attributes, where each choice will result in different concept lattices. For academic papers, the natural choice is to use the papers themselves as objects in the context and to assign all additional information such as the paper's authors, the publication year, and the extracted keyphrases as attributes in the context.

Attributes can be multi-faceted and in our approach we distinguish between various categories of attributes. For example, it is possible for the same value (e.g. 2000) to appear twice in an attribute set, once representing a year and once representing a keyphrase.

Figure 4 shows an example of a context table in which papers are objects with associated attributes. Each attribute is indicated as a column in the table, while the papers form the rows in the context table. An “X” in the context table indicates that an attribute applies to the corresponding paper. Many algorithms exist that directly convert a formal context into a concept lattice (e.g., Lindig (2000); Ganter (2010)).



Paper_ID	Alice	Ben	Theorem Provers	Formal Concept Analysis	2000	2005	2015
paper-1	X	X	X	X	X		
paper-2		X		X	X		
paper-3	X		X			X	
paper-4	X			X			X
paper-5		X		X			

Fig. 4: An overview of our navigation approach using formal concept analysis. We construct a formal context table from paper meta-data and extracted keyphrases which is then used to automatically generate a concept lattice for the publication data. The initial focus is the top concept in the lattice which generates a tag cloud for the full dataset. As tags are selected the focus moves down in the lattice and the tag clouds are refined.

The constructed concept lattice allows us to easily identify which papers share particular attributes. However, when the dataset becomes large the Hasse diagram of the concept lattice becomes very difficult to interpret. An image of a large concept lattice of academic publication data can be observed in (Liu et al. 2015, p. 12) which shows that concept lattices become uninterpretable when they become too large. We therefore use tag clouds, constructed from concept lattices, to overcome this limitation.

3.2 Tag Clouds

Tag clouds (also called word clouds) are a common text visualization method. Tag clouds can be generated from user-generated tags of a particular content or directly from tokenized texts. Tags are usually sized according to the frequency of their occurrence in the text and can be color-coded to reflect additional information.

In our approach, the tag clouds are generated directly from concepts in the underlying concept lattice. All attributes of the defining concept of each object in the extent of the focus concept (each academic paper) are collected and added to a multiset which forms the tag cloud (Greene and Fischer 2014, 2015).

Definition 7 *The* tag cloud *from a concept* $c = (O, A) \in B(\mathcal{C})$ *is defined as* $\tau(c) = O \uplus \biguplus_{o \in O} \pi_A \sigma(o)$.

The more frequent the occurrence of an attribute is (i.e., the more papers it applies to) the larger its tag is displayed in the tag cloud. Therefore, tags in the tag cloud are sized according to how well they characterize the current set of papers (i.e., the extent of the current focus).

The simplest and most popular tag cloud layout (Lohmann et al. 2009) is as an alphabetically sorted list of tags in a roughly rectangular shape which was found by Schrammel et al. (2009) to perform better than random or semantic layouts. We use this layout because it simplifies textual search within the tag cloud. Each tag i is scaled between the given minimum and maximum font sizes f_{min} and f_{max}, according to its weight t_i in relation to the minimum and maximum weights in the context table, t_{min} and t_{max}; hence,

$$\text{size}(i) = \left\lceil \frac{(f_{max} - f_{min}) \cdot (t_i - t_{min})}{t_{max} - t_{min}} \right\rceil + f_{min} - 1$$

for $t_i > t_{min}$ and $\text{size}(i) = f_{min}$ otherwise.

3.3 Navigating Concept Lattices Using Tag Clouds

Navigation in the tag cloud is supported through the underlying concept lattice. When a tag is selected or de-selected, navigation in the underlying lattice is triggered and the tag cloud is reconstructed from the updated focus concept.

In our approach we maintain a *focus concept*, from which the current tag cloud is rendered as described above. When the user selects (or deselects) a tag, the browser computes the new focus concept and subsequently re-renders the tag cloud. The focus, or more precisely, its extent contains the subset of objects in the dataset that share all currently selected tags. Therefore the set of currently selected tags can be regarded as the current query. The initial focus (corresponding to an empty selection set) is therefore the lattice's top element, whose extent contains the entire publication set. Therefore, the initial tag cloud contains aggregated information for the full dataset of publications.

Navigation is refinement-based such that when the user selects an additional tag, the browser updates the focus by computing the meet of that tag's defining concept and the previous focus. Intuitively, deselection should be the inverse of selection and therefore deselecting the last selected tag should move the focus back to its previous position. The focus is recomputed as the meet of the defining concepts of the remaining selected tags, in order to provide a de-selection operation which is the inverse of the selection operation. Note that not only the most recently selected tag is available for de-selection, and therefore the de-selection of a tag does not only function as an undo operation.

Figure 4 shows an overview of the navigation process. After the context table has been constructed from the publication data the concept lattice is generated. The focus in the lattice is initially the top concept, which contains all papers in its extent. Therefore, the initial tag cloud displays the aggregated attributes from the full publication dataset, sized according to how many papers the attributes apply to. When a tag is selected in the tag cloud, the focus in the lattice is updated to the meet of the defining concept of the new selection and the previous focus. A new tag cloud is then generated from the updated focus.

3.4 Object Relationships in the Tag Cloud

Concept lattices do not directly provide a mechanism for handling relationships between objects in a single lattice such as the reference and citation relationships between academic papers. We overcome this drawback by generating additional concept lattices on the fly to present the references and citations for a particular paper or attribute (such as an author or keyphrase) in an additional tag cloud.

The most obvious solution for relating different objects in the context table is to add references to other objects to the attribute set of a single object. In that way academic papers would be used as both objects and attributes (i.e., whenever they are referenced or cited) in the context table. The category of the attribute could be used to indicate whether the relationship between the two papers is a reference (i.e., an outgoing link) or citation (i.e., an incoming link). However, following this approach of selecting a paper in the tag cloud would show only tags for the papers (represented by their paper-ids) that form references and citations for the selected paper. No additional information about the references and citations (such as authors, keyphrases or publication years) would be present in the tag cloud and it would be necessary to select the tag (the paper-id) for each individual referenced and citing paper to gather any information about the papers themselves. Therefore, while using referenced and citing papers as attributes in the context table is possible, this does not provide any additional information about the referenced and citing papers and is very limited in usefulness in terms of the navigation options provided.

We instead generate additional concept lattices for the references and citations for a particular paper or any attribute in the tag cloud on the fly so that we can present aggregated citation and reference information to the user. Any tag in the tag cloud can be used to construct an additional tag cloud of the references or citations for that particular attribute. Therefore, it is possible to generate a tag cloud which displays aggregated information of the references or citations for an individual paper, but it is also possible to generate a tag cloud which displays aggregated information of papers that have cited a particular author, or papers that have cited papers that apply to a particular keyphrase.

Note that the object types in our citation and reference lattices are no longer paper-ids but representations of each citation so that we fully exploit the citation data and record the frequency of each citation. For example, when a paper is referenced more than once by a collection of papers (e.g., all papers by a particular author) this paper will appear as an object in the reference lattice multiple times. In this way our reference and citation tag clouds not only reflect the citation or reference but also its importance which is represented by the size of the tag.

When an additional tag cloud of references or citations is created, we create a new concept lattice which contains only the relevant subset of academic papers as objects. We then present a tag cloud corresponding to the newly created concept lattice and allow users to navigate in this tag cloud as well, in order to browse the references and citations of a paper or attribute.

For example, selecting the keyphrase "Model Checking", as can be seen in the top image in Figure 1, and generating the the two additional tag clouds for references (bottom left) and citations (bottom right), provides the user not only with an overview of papers associated with "Model Checking", but also papers that either cited "Model Checking" papers or are referenced by "Model Checking" papers. From these tag clouds we see what other keyphrases are associated with "Model Checking", as well as which authors have provided the foundational (referenced work) for the field of "Model Checking". We can restrict the tag cloud to show which authors, and over which time period, have published papers referencing "Model Checking" papers. Similarly, the reference and citation tag clouds can also let researchers discover authors that they have frequently referenced, as well as which authors are referencing their work.

3.5 Tool Support

We built a customized publication browser on top of our generic ConceptCloud framework (Greene and Fischer 2014, 2015) suited for handling academic publication data. Our ConceptCloud browser framework allows users to generate a concept lattice and tag cloud visualization for semi-structured data collections that can be provided as JSON or XML input. The browser provides various preprocessing steps (such as stemming and stop word removal) which can be applied to data sets that are input. The browser can be used to generate a concept lattice from a dataset and since the concept lattice is a generic structure, the tag cloud visualization can be then used to present the information in any concept lattice and support navigation. Our browser framework has previously been applied to software development repository data (Greene and Fischer 2014, 2015, 2016) but is generic enough to be applied to any semi-structured data collection.

Tags in the tag clouds generated by ConceptCloud are colored according to the type of information that they represent (e.g. author tags are orange). Hovering over a tag will display its category as well as its count. The tag clouds can also be filtered to include only tags of certain sizes, as well as to include only specific categories.

Figure 5 shows the initial interface for our publication browser. The user can start exploring the information displayed in the tag clouds without first having to come up with any search terms. When a tag is selected it will be indicated in the cloud as red, so that users can easily keep track of their tag selections.

The search functionality provided in this adaption of the ConceptCloud is an extension of selecting a tag. In other words, suggestions that match the search query are tags, colored according to their tag category, from which the user has

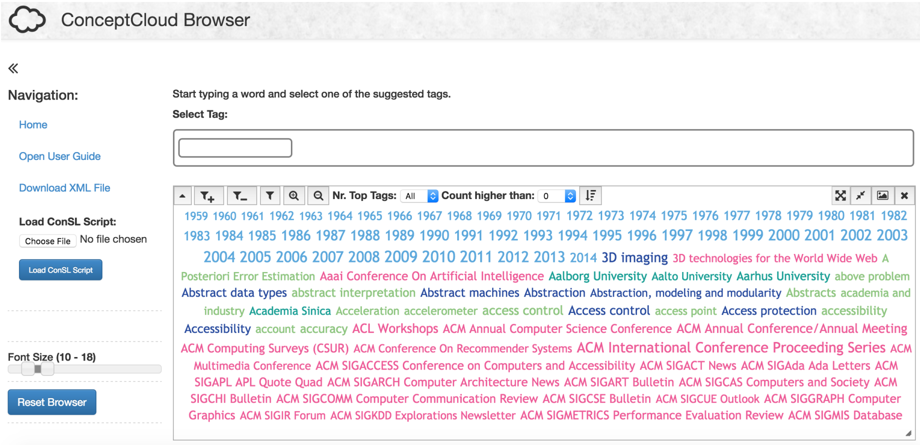


Fig. 5: Overview of the initial interface for ConceptCloud publication browsing. A search feature is provided to be able to easily identify and select tags that are selectable in the current cloud.

to select an appropriate tag. Multiple searches are possible by further filtering the current tags. In Figure 6, for example, the second search only searches through tags applicable to the current selection of "Model Checking". Therefore, two consecutive searches provide the intersection and not the union of the two tags.

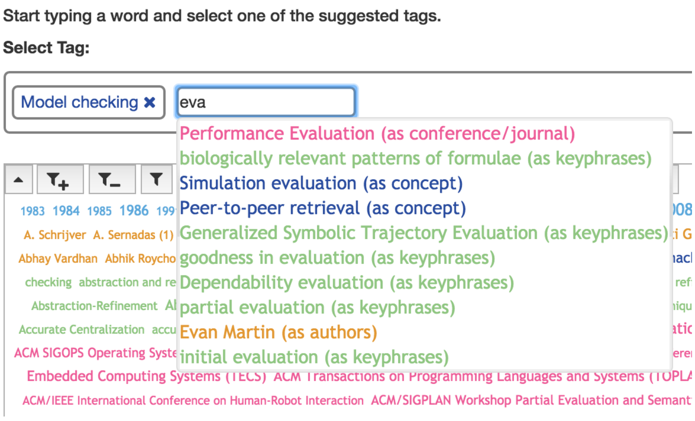


Fig. 6: Tag suggestions displayed on start of typing a tag name. Tags in the suggestion list are colored according to the tag category they belong to in the tag cloud.

Furthermore, all the papers in the current focus's extent are listed in a table below the tag cloud as shown in Figure 7. Therefore, users are able to see a traditional list of papers (including titles and abstracts) to which the current tag selections apply and updates each time a tag is selected or deselected.

Title	Year	Abstract	Journal/Conference	Authors	Key Phrases	Concepts	Citation Count
Test input generation with java PathFinder	2004	We show how model checking and symbolic execution can be used to generate test inputs to achieve structural coverage of code that manipulates complex data structures. We focus on o...	International Symposium on Software Testing and Analysis	Corina S. Păsăreanu (1), Sarfraz Khurshid, Willem Visser	white-box test input generation, black-box fashion, structural coverage, Java PathFinder model checker, test input generation, coverage, testing object-oriented programs, red-black tree implementation, red-black trees, Java TreeMap library, model checking, symbolic execution, straight model checking	Software creation and management, Abstract machines, Software verification and validation, Symbolic and algebraic manipulation, Theory of computation, Semantics and reasoning, Program reasoning, Software and its engineering, Computing methodologies, Abstraction, Models of computation	137
					state-explosion problem, possible program behaviors, Concurrency, Iterative		

Fig. 7: Table view for "Model Checking" selection in ConceptCloud. The papers are automatically sorted in descending order of citation count and provide information not present in the tag cloud, such as the paper's title and abstract.

3.5.1 Linked Views

Categories can also be separated into their own tag cloud views, to provide a multi-faceted view. All the viewers are linked such that when a selection is made in any of the tag clouds, all of the linked clouds are updated with the new selection. Figure 8 shows an example in which different categories (keyphrases, authors and years) have been separated into their own individual viewers.

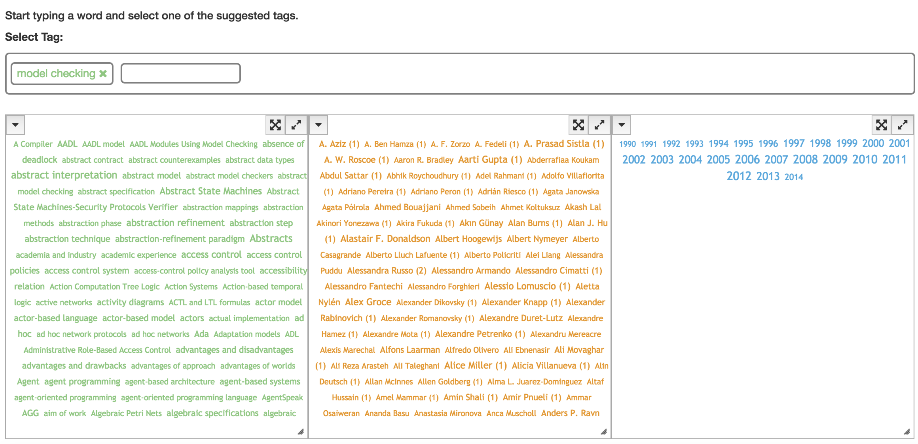


Fig. 8: Category viewers in ConceptCloud. From left to right, keyphrases, authors and years all applying to the selection of the keyphrase "Model Checking". Viewers are linked so that when a selection is made all viewers will update according to the new selection.

A viewer can also be constructed with its own unique tag selection that applies only to that viewer; these are called "sticky tags". Any tag can be used to construct a sticky tag viewer, which will then be presented in a new viewer window where all the selections of the main cloud will also be applied along with the selection of the sticky tag. While additional tags can be selected and de-selected in all viewers, the sticky tag can never be de-selected from its sticky tag viewer. Sticky tags enable the selection of tags which would otherwise not be selectable at the same time, such as years which cannot be selected together as each paper is only published in a single year. Sticky tags also allow the comparison of different tag selections. Figure 9 shows an example of sticky tag viewers for two years. The global selections still apply to the sticky tag viewers but the sticky tags apply only to the particular viewer.

Note that sticky tag viewers are not the same as the viewers used to represent the citation and reference clouds, as they are still derived from the main concept lattice whereas the citation and reference clouds are each derived from their own new lattices which are generated on the fly. In the browser interface all viewers look the same, however, sticky tag viewers are labeled according to their sticky tags and citation and reference clouds are labeled according to the paper or property used to generate them (e.g., the keyphrase "model checking").

Fig. 9: Sticky tag viewers in ConceptCloud enabling the comparison of "Model Checking" papers in different years.

4 User Study Methodology

We performed a user study to evaluate whether users understand and are able to make use of all the functionalities provided by our publication browser. Participants were asked a wide variety of scientometric questions and evaluated on whether they can find and retrieve the correct information, such as papers related to a particular topic of interest, prominent researchers in a field, and journals and conferences best suited to different research topics. For this user study we used the ACM DL dataset to populate the publication browser with 1 848 048 papers of which 1 399 805 have abstracts that are necessary for the keyphrase extraction. Therefore, the user study was set up primarily to evaluate the usability of the publication browser. Secondarily, in context of the ACM DL dataset, the scientometric questions served to test whether the participants were able to use the publication browser's functionalities to retrieve the correct information and whether they interpret the visualized information correctly.

It should be noted that since we used the ACM DL dataset to populate the publication browser, it only contains citation and reference information from the closed citation network of the ACM DL. Therefore, the citation counts displayed in our browser may differ from other indexing services and the correctness of the participants' answers in the user study are judged strictly in the context of the ACM DL data. However, our browser is flexible enough to be used with any other academic publication datasets and new data sources could simply be added by merging paper and author entities, should new data sources become available.

We grouped the publication browser's functionalities into eight main categories, such as selecting tags and creating viewers, and formulated the questions so that participants are required to use all of the tool's functionalities to answer the questions successfully. The list of function categories with the number of questions requiring the corresponding functionalities is given in Table 1.

	Functionality	**Nr. of Questions**
F1	Selecting/Deselecting Tags	17
F2	Searching for tags using the provided search functionality	12
F3	Filtering categories and creating category-specific viewers	18
F4	Filtering viewers by number or size of tags	15
F5	Sorting the tags according to tag count	15
F6	Creating viewers with "Sticky Tags"	2
F7	Creating Citation and Reference clouds	4
F8	Interpreting the table view	3

Table 1: Main functionalities provided by the publication browser grouped into categories. The second column shows the number of questions in the user study that require the associated browser functionality to answer the question.

It should be noted that the number of questions associated with the various functionality categories differ substantially. This is due to the fact that certain functions are prerequisites for others. For example, selecting or searching for a tag is required before creating a citation or reference cloud. Rather than forcing an equal number of questions for each functionality, the distribution more closely reflects a real-world use case.

Similar to the study conducted in Osborne et al. (2013), we performed a non-comparative study since most academic indexing platforms and digital libraries do not provide the required functionality to answer the type of complex questions that our publication browser has been designed to answer. Therefore, a comparative study between two or more tools would unfairly favor one tool over the others depending on how complex the questions would have been constructed.

Participants first watched an introductory video[1] about the tool to familiarize them with the usage of the tool. Each user study was conducted separately and the authors observed each participant and made notes of exactly which tool functionalities they made use of and how long they took to answer each question. We used a stopwatch to time the participants on each question and did not include the time they required to write down their answers on paper. We applied a total time limit of 90 minutes per participant to complete the study. We checked that answering all questions was feasible with this time limit and confirmed it through the pilot and the actual user study.

4.1 Question Set

We constructed 22 questions that cover various topics in scientometrics to establish whether users are able to answer common scientometric questions using the publication browser. The questions cover eight broadly defined topics such as citation analysis (de Solla Price 1965; Garfield 1979), author collaboration (Abt 2007; Wallace et al. 2012), author ranking (West et al. 2013; Hirsch 2005), topic trends (Rosvall and Bergstrom 2010), and university rankings (Eccles 2002; Aguillo et al. 2010). The complete list of topics covered by the questions is given in Table 2.

	Topic	**Nr. of Questions**
T1	Citation analysis	6
T2	Author ranking	2
T3	Author activity	4
T4	Author collaboration	5
T5	Topic trends	3
T6	University ranking	2

Table 2: The number of questions for each topic proposed to participants in the user study.

We have formulated some of the questions in our study more generally to evaluate how the participants would make use of the tool in order to answer scientometric questions according to their own interpretation. For example, finding a prominent author depends on the participant's interpretation of whether citation counts or the number of publications are used as a metric to define "prominent". We have phrased question Q1 as "Who is the most prominent author in the field of Machine Learning?" rather than directly asking participants to select the tag for the keyphrase "Machine Learning" and evaluate which author has the largest tag size in the tag cloud (i.e., the most publications on this topic) as that would only measure the participant's ability to select tags and follow instructions and not their ability to answer scientometric questions with the tool. As a result of the question phrasing, there are multiple "correct" answers for some of the questions, and during the study we recorded how the participants made use of the tool to answer the questions.

[1] The introductory video of the ConceptCloud Browser for academic papers is available at `https://www.youtube.com/watch?v=8zJ618yOWBI`.

Table 3 lists the questions that the participants had to answer during the user study. The table indicates the category in which each question falls (T1-T6), as well as the tool functionalities (F1-F8) that need to be used to answer the question successfully. However, in the user study the participants were only provided with the questions without any additional hints, such as the topic of the question or the required tool functionalities. We also randomized the order of the questions for each participant to allow us to identify any learning effects during the course of the participants' usage of the tool. It should be noted that the names of authors, series and years are arbitrarily selected, and replacing these variables with other values would still require participants to answer the questions in the same way.

Osborne et al. (2013) performed a non-comparative user study in which they posed four tasks (including a warm-up task) to participants. One task that had to be completed was formulated as follows: "Find the top 5 authors with the highest number of publications in *Semantic Web* and rank them in terms of number of publications in *Artificial Intelligence*. For each of them find their most cited paper." Their study is only based on the fields of "Semantic Web" and "Artificial Intelligence" and they do not use a cross-disciplinary library of publications, such as the one our study is based on. Therefore, our questions are formulated less restrictively and are closer to real-world questions that researchers would ask.

Dunne et al. (2012) also performed a non-comparative user study for which they recruited participants that were researchers in computational linguistics, which is the same field as the papers in the dataset they used for their tool. The study conducted in (Dunne et al. 2012) asked participants to "(a) Identify and make note of important authors and papers and (b) find an important paper and collect evidence to determine why it is important". We posed a similar question in our study which also asks the participants to identify the important authors in a field (Q1). However, we have extended our question set to cover a wide variety of scientometric topics that are of interest in the literature. Alternative evaluation approaches have included deploying the evaluated tool and collecting usage data (Dörk et al. 2012). While we have not followed this approach, we have observed and taken note of each participant's usage of our tool individually to identify whether they were able to use the functionalities successfully.

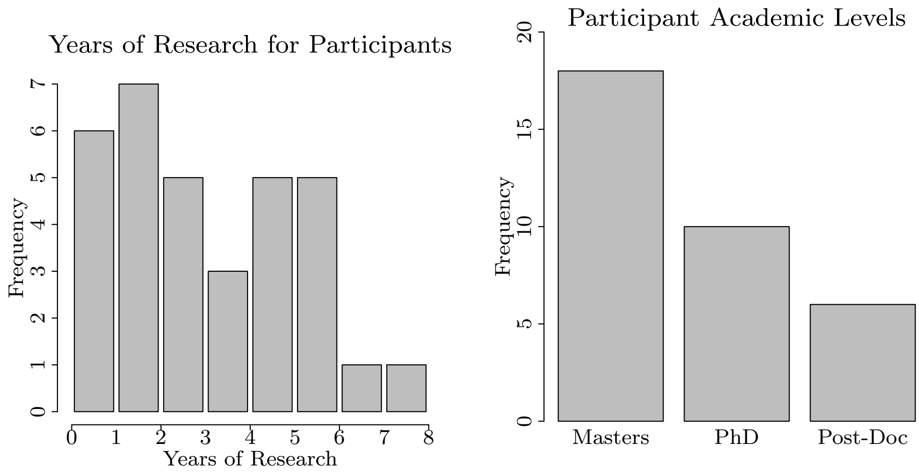
4.2 Population

We asked post-graduate students and post-doc researchers from various departments at Stellenbosch University to participate in our user study. Participation in the study was voluntary and participants were not compensated in any way. A total of 39 participants from various academic fields took part in the user study.

We performed a pilot user study with five participants after which we made changes to the tool and the question set. The results shown in this paper are obtained from conducting the user study with the remaining 34 participants. Figure 10a shows the distribution of postgraduate research experience in years for the 34 participants, while Figure 10b indicates their academic ranks.

	Question	Functionalities	Topic
Q1	Who is the most prominent author in field of “Machine Learning”?	F1/F2, F3, F4/F5	T2
Q2 (a)	Which country does “Germany” collaborate with the most?	F1/F2, F3, F4/F5	T4
Q2 (b)	In which year did authors from these countries publish the first collaboration paper?	F1, (F3)	T4
Q2 (c)	How many authors collaborated in this year from these two countries?	F1, F3	T4
Q3	Which are the top 5 affiliations according to publication counts in 2010?	F1/F2, F3, F4/F5	T6
Q4 (a)	List the three main topics author “Ivan Bratko” works on.	F1/F2, F3, F4/F5	T3
Q4 (b)	Create a viewer showing the years he published papers about his most prominent research topic. Show the viewer to the observer of the study.	F1/F2, F3	T3
Q4 (c)	In which years did he publish the most papers overall?	F1, F4/F5	T3
Q4 (d)	With whom does he co-author the most papers?	F3, F4/F5	T4
Q4 (e)	Name the author who cites him the most.	F3, F4/F5, F7	T4
Q5	What are the 3 main keyphrases of papers that cite papers with the keyphrase“Machine Learning”?	F1/F2, F3, F4/F5, F7	T1
Q6 (a)	In what year were the first papers published about “Internet of Things”?	F1/F2, (F3)	T5
Q6(b)	If you had a paper about “Internet of Things”, in which journal/conference would you publish if you’re only interested in being cited as often as possible?	F3, F4/F5, F7	T1
Q7 (a)	Compare the topics in the first and last year of the series “International World Wide Web Conference”. How have they changed over time?	F1/F2, F3, F4/F5, F6	T5
Q7 (b)	What is the title of the paper with the most citations in this conference?	F8	T1
Q7 (c)	Who published the most papers in the first year of this conference?	F1, F3, F4/F5	T2
Q7 (d)	Where did this author publish most papers?	F1, F3, F4/F5	T3
Q7 (e)	In which journal/conference is this author’s most cited paper?	F1, F2, F8	T1
Q8	What are the most prominent keyphrases covered by the “International Conference on Software Engineering” and how do these differ from the conference “Foundations of Software Engineering”?	F1/F2, F3, F4/F5, F6	T5
Q9 (a)	Which is the most cited “International Conference on Software Engineering” paper and what is its topic?	F1/F2, F8	T1
Q9 (b)	Which journal/conference is most referenced in “International Conference on Software Engineering”?	F3, F4/F5, F7	T1
Q10	If you had to advise “Ben Carterette” on choosing the most relevant university to collaborate with, which university would you suggest and why?	F1, F2, F3, F4/F5	T6

Table 3: The questions asked in the user study. Each answer requires the use of one or more functionalities of the publication browser, given in the second column. Some functionalities are optional, indicated with round braces, while sometimes two different functions can be used to get to the correct answer (indicated with ‘/’). The third column gives the topic associated with the question.



(a) Years of research experience of participants.

(b) Participants' academic ranks.

Fig. 10: Population characteristics.

4.3 Pilot Study

Initially, we performed a pilot study with five participants after which we re-evaluated the user study setup before continuing. The tool initially included a traditional search interface which allowed users to search for tags by submitting free-text queries. In addition, all questions were self-contained without any continuation from one question to another. The combination of these two factors led the users to approach the tool as a traditional search tool because of the familiarity with the concept of search.

After noticing the heavy reliance on the search functionality, we adapted the question set slightly to encompass more continuation between questions. For example, we changed the follow-up question of "List the three main topics author "Ivan Bratko" works on", from "In which years did "Ivan Bratko" publish most papers overall" to "In which years did he publish the most papers overall". Furthermore, we added additional follow-up questions. For example, we added questions Q2 (b) and Q2 (a) to question Q2 (see Table 3), to encourage users to keep current tag selections and to engage in browsing and exploratory search (White and Roth 2009) instead of using the search functionality at the beginning of each question.

We also changed the search interface so that participants could no longer search using free-text queries but instead would have to select a tag from a list of suggestions that would appear after they started typing as shown in Figure 6. Furthermore, using the associated color for the suggestions according to the color of the tag categories in the tag cloud would help users realized that the search function was simply another mechanism of selecting a tag, instead of manually looking for a specific tag in the tag cloud.

In the pilot study we provided participants with only a user manual detailing the tool's features, however, since not all participants observed the manual, we showed an introductory video describing the use and functionalities of the tool to the participants before beginning the user study.

5 User Study Results

In the user study we collected detailed information about each participant's usage of the tool. For each question we recorded the time taken to complete the question, the functionalities that the participant used to answer the question, as well as, the correctness of the answer according to the participant's interpretation of the question. Note that each participant received the questions in a random order.

5.1 Time Taken per Question

To evaluate whether there was an observable learning effect for participants we plot the average time taken for questions in question order (Figure 11a) and the average time taken for questions in the order that the participants received them (Figure 11b). For example, Figure 11b shows the average time taken by the participants to answer their first questions despite the fact that the actual question they were answering was random. Figure 11c shows the normalized time for each participants' questions in the order that they received them. More precisely, the normalized time for participant A's first question (Q7) is the time they required to answer Q7 minus the average time that all participants required for answering Q7.

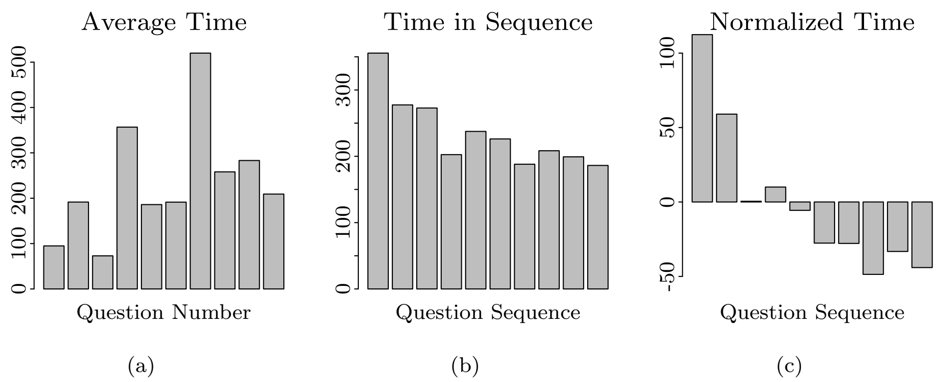
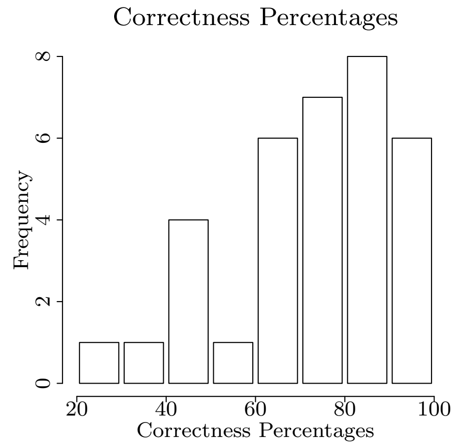


Fig. 11: Average time taken per question in (a) question order (b) the order that the participants received them (c) the order that the participants received them normalized by the average time of all participants per question.

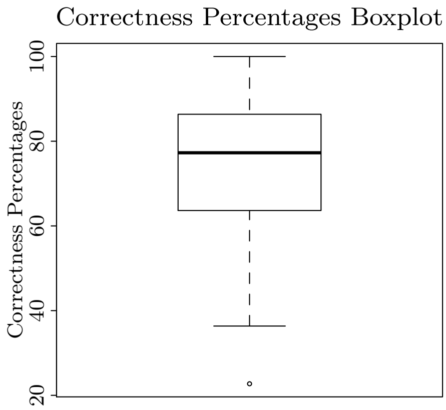
A comparison between Figures 11a and 11b shows that the patterns of question timings differ substantially when the average time was calculated according to question number or question sequence. From 11a we can see that participants took the longest time to answer question Q7 followed by Q4, both of which comprise of multiple subquestions. Figure 11b shows that, on average, participants took the longest to answer their first question, regardless of what question this was, and the time decreased as participants answered more questions. Figure 11c confirms this learning effect and shows that when a participant received their first question they took more than 100 seconds longer to answer it than the average time needed on that question by all participants. By the third question the participants were able to answer it as fast as the average time required for that particular question. On average, the participants started with their third question after 10 minutes and 55 seconds. After the fourth question the participants were able to answer all questions faster than the average time taken by all participants to answer the respective questions, which corresponds to 18 minutes and 47 seconds until effective usage of the tool.

5.2 Correctness Percentages

We also analyzed the participants' answers for correct results to compute correctness percentages over all their questions. Figure 12a shows the participants' percentages of correct answers from which one can see that most participants answered more than 60% of the question correctly. Figure 12b shows a box and whisker diagram for the correctness percentages achieved which shows that the highest score achieved was 100% with a median score of 78%.



(a) Histogram of Correctness Percentages Achieved



(b) Boxplot of Correctness Percentages Achieved

Fig. 12: Correctness Percentages Achieved

We performed the same analysis as in Figure 11 using the correctness scores instead of the timing information. Figure 13b shows that more participants got their first question incorrect, regardless of which question they received first. However, as can also be seen in Figure 13c, after the first question there is no other indication that the sequence of the question affected the correctness of the participants answers.

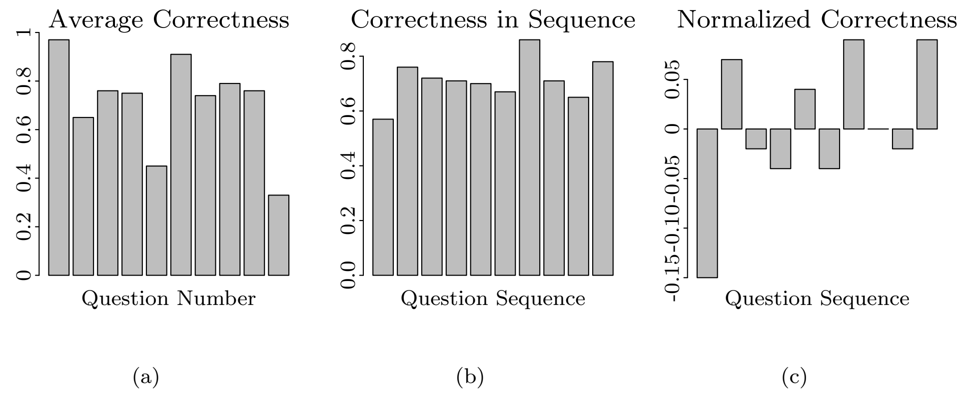


Fig. 13: Average correctness for questions in (a) question order (b) the sequence that participants received them (c) the sequence that participants received them normlized by the average correctness that all participants obtained for the corresponding questions.

5.3 Results per Question

Figure 14 shows a box and whisker diagram for each individual question showing the variation in times required per question. In the heading of each box and whisker diagram we indicate the average percentages that were achieved by all participants and whether the corresponding question required making use of a more complex tool functionality (F6 - use of a sticky tag viewer, F7 - use of a citation or reference clouds, F8 - use of the table view) to answer the question.

From Figure 14 we can see that no questions are outliers, in terms of time taken to answer the question, and that there is no identifiable pattern between the use of an advanced tool functionality and the participants' completion time. From the results of the individual questions, we see that Q5 and Q10 were the only questions in which the participants obtained an average correctness percentage below 50%. While question Q5 required the construction of a citation cloud, question Q10 only required straight-forward tool functionalities but the question itself was more complex and required the participants to suggest a relevant university for a particular author to collaborate with. Note that some questions (e.g., Q9 and Q10) have outliers in terms of the average completion time, which could be due to the random order in which participants received the questions.

5.4 Participant Feedback

After each participant completed the user study we asked them to fill out an anonymous feedback form about their experience using the tool. We asked partic-

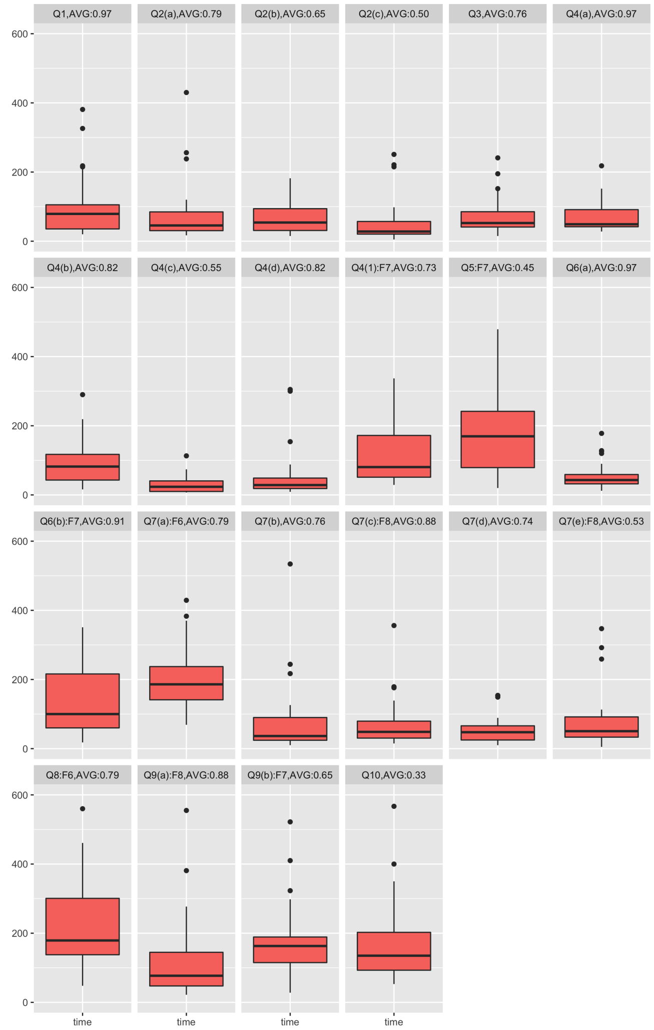


Fig. 14: Boxplots per individual question. The question number is indicated in the heading for each boxplot along with the average correctness achieved for the question. The heading of each boxplot also indicates whether the question required any of the advanced functionalities sticky tag viewers (F6), citation or reference clouds (F7) or the table view (F8).

ipants to rate the different tool features on a Likert scale (1-5) according to how useful the participant found them and how easy the participant found the feature to use. In addition, the participants were asked to give feedback on any areas of the tool that could be improved and provide any additional comments about their experience using the tool.

Table 4 shows the mean and median scores for each tool feature. While participants found selecting tags and searching for tags both very useful and easy to use, the feedback for filtering the tag cloud views to show only certain categories was indicated as very useful but not as easy to use. Filtering the number of tags shown in a particular view was indicated as less useful but still easy to use. However, the sticky tag views were shown to be less useful and more difficult to use. The citation and reference clouds were indicated as the most difficult features to use but were rated as more useful than both the filtering of the tag clouds according to tag category and tag count. Participants also scored the tool with a mean of 4.49, when asked to indicate how likely they would use the tool in their own research.

Feature	**Mean Usefulness**	**Median Usefulness**	**Mean Ease of Use**	**Median Ease of Use**
(F1) Selecting/Deselecting Tags	4.27	5	4.64	5
(F2) Searching for Tags	4.86	5	4.46	5
(F3) Filtering Categories and creating category views	4.46	5	4.00	4
(F4) Filtering the number of tags	3.55	4	4.41	5
(F6) Sticky Tags	3.59	4	3.91	4
(F7) Citation and reference clouds	4.18	4	3.82	4

Table 4: Mean and median of participant scores for the usefulness and usability of tool features.

We used a process of open coding on the free-text feedback answers provided by participants to the questions "Are their any areas of the tools you feel could be improved" and "Please provide any comments on your experience of the tool". We identified the major themes of the participants responses and noted that out of the 32 participants who were willing to provide anonymous feedback on the tool the themes identified were as follows.

47% of participants felt that some additional functionality would have improved their experience of the tool. For example one participant noted that they would like to "[be] able to create a new viewer from a selected tag. [They] could search for a tag, but [it] was unclear on how to make that tag a new viewer". 40 % of participants indicated that they had difficulty understanding at least one feature of the tool. These included features such as where the tag counts were derived from (citation count or publication count) and the effect of filtering the number of tags.

25% percent of participants indicated that at least one function of the tool did not behave as expected and 25% also indicated that they experienced a definite learning curve in their use of the tool. For example, one participant noted that "Once [they] got the hang of it, [they] found it very user friendly and can imagine it being very useful for researchers...".

Participants labeled the tool as useful (38%), easy to use (19%), powerful (19%) and fun to use (16%). For example, participants noted that "[t]his could come in as a very useful tool in the future". There were also some participants that reported feeling impartial to the tool (6%) and found it complicated (6%).

6 Analysis of Results

6.1 Effect of participant's background on correctness results achieved

We performed one-way ANOVA tests to determine whether a participant's background (years of research and academic rank) had any effect on the time required to answer all questions or the correctness percentage they achieved. We first performed a Shapiro-Wilk test on the participants timing information to confirm that the timing information was normally distributed and therefore appropriate for an ANOVA test.

When grouping the participants according to their years of research experience we found that the number of years of research experience did not make a statistically significant difference to the time it took the participants to answer the questions (p-value 0.937). When grouping participants into their academic ranks of Masters, PhD and Post-Doc we found that the academic rank also does not have a statistically significant effect on the time taken to answer questions (p-value 0.306).

We also tested whether the participants' academic ranks and years of research experience made a statistically significant difference to their correctness percentages obtained. However, since the correctness percentages were not normally distributed (according to the Shapiro-Wilk test) we performed a Kruskal-Wallis test instead of an ANOVA test. We found that the correctness percentages were not statistically significantly different for any of the academic ranks of the participants as we could not reject the null hypothesis (p-value 0.123). We also tested whether the participants' years of research affected their correctness percentages and again could not reject the null hypothesis (p-value 0.129) indicating no statistically significant difference between the correctness results of participants with different years of research.

6.2 Effect of use of a specific tool function on correctness results achieved

We also evaluated whether the required use of a specific more complex tool feature influences the correctness percentage achieved.

We tested this effect using a Pearson Chi-Square test since, for each question, we had categorical data stating whether or not a specific tool function was required. We found that for all questions requiring more complex tool functions F6 (sticky tags), F7 (citation and reference clouds) and F8 (use of the table view) we could not reject the null hypothesis that there is no association between the presence of the tool function and the correctness achieved. For the questions requiring F6 vs questions not requiring F6 we obtained a p-value of 0.27, for F7 we obtained a p-value of 0.1 and for F8 we obtained a p-value of 1. Therefore, we see that the involved tool function does not affect the correctness results.

6.3 Threats to Validity

6.3.1 Internal Validity

There are various uncontrolled factors that might have influenced the results of our user study.

Computer Equipment: The user study was conducted using a centralized lab server and so it is possible that some participants experienced slower loading times than others. However, since there were no time limits to specific questions in this study this is unlikely to have affected the user study responses.

Participants: To make sure all the subjects were competent enough to take part in our study we only used participants at the post-graduate level who would have been familiar with the concept of references and citations.

Training: The effect of the training that we provided to participants might have influenced the experimental results. However, since we provided a user manual and let the participants watch an introductory video the same training was provided to all participants.

Tasks: The choice of tasks might have been biased towards the types of questions that can be answered using out ConceptCloud browser. To mitigate this risk, our tasks were formulated using common scientometric questions as a baseline so that the tasks provided were real-world questions.

6.3.2 External Validity

There are a few factors which may influence the generalizability of our user study results.

Participants: The participants involved in the user study might not be representative of a real-world sample of academic researchers. However, all participants were pursuing post-graduate degrees or were post-doctorate researchers at the time of the study. We also have participants of varying levels of research experience (from 1-8 years of research experience) which did not have a statistically significant impact on the correctness achieved or the time taken by the participants.

Marking of Question Answers: Answers to questions in the user study were marked by the authors, which can introduce subjectiveness, especially since some questions were stated vaguely and multiple correct answers existed. For example, the interpretation of “prominent author” will inevitably vary between participants. However, the participants answers were only marked as correct, if their understanding of “prominent author” was scientometrically plausible.

Participant Feedback: The results from the feedback of participants described in Section 5.4 might contain response biases. Participants were aware that the tool had been developed at their home university which may have influenced them to give the tool higher scores. However, since we were not directly comparing our tool to another tool their is no risk of the participants unfairly favoring our tool over another. We also told each participant that their feedback form was anonymous before they filled out the form.

7 Related Work

In this section we discuss closely related work on approaches and tools that support the browsing of academic publication data as these can be best contrasted to our approach. Background on keyphrase extraction and formal concept analysis is described in Sections 2 and 3.1, respectively.

SurVis (Beck et al. 2016) allows users to construct an interactive visualization of paper collections that appear in literature survey papers. SurVis also provides a word cloud visualization for papers contained in a literature survey where different types of meta-information are presented in separate word clouds. Our publication browser also allows different categories to be separated into their own tag clouds as well as providing the option to have all tags in the same tag cloud, differentiated by color.

Medlar et al. (2016) developed the PULP system for exploratory search of scientific literature. PULP presents information about topics covered in graphical format, where topics for each year are listed and lines between the different years are used to indicate how the topics over the different years are related. The user can select topics in the visualization to show keywords associated with the topic and once these selections have been made a search interface is used to display academic papers related to the selections. The user then provides relevance feedback on the set of documents that has initially been provided and can then load more documents which will be based on which of the previous documents were marked as relevant. PULP also supports an exploratory search approach and allows users to select topics from a visualization as we do in our approach as well. However, we also allow users to select other fields of interest such as authors or affiliations. We also present the titles of the papers in a list format but present the full list to the user and allow them to continue refining the list by making additional tag selections.

The ASE tool (Dunne et al. 2012) allows users to familiarize themselves with a new research field. ASE operates on "citation text" which details citing papers that refer to the paper's content compared to our approach of using text extracted from the papers' titles and abstracts. ASE presents multiple inter-linked views (as can also be constructed in ConceptCloud) showing citation network visualizations as well as citation numbers for a specific group of papers. An in-cite summary which lists text surrounding a citation is also present in ASE to provide an overview of a paper. ASE operates on a smaller subset of data compared to the ACM DL dataset and therefore the goals of the analysis are somewhat different. The tool allows users to explore a subset of papers that have been retrieved through other means (search) as opposed to our approach which functions as an interface for discovering new fields and papers in the broader library.

Dörk et al. (2012) use a "strolling" technique which allows users to explore publication data. Instead of refinements the PivotPaths tool uses pivots to change the information currently displayed in the browser, to allow users to not only explore a dataset by making refinements but also by navigating laterally in the dataset. Using our ConceptCloud browser, tags in the tag cloud can be de-selected in any order (not simply as an undo operation) which facilitates lateral navigation in the concept lattice, and therefore also in the underlying dataset.

Osborne et al. (2013) do not make use of keyphrases but instead use the OWL ontology supplied by the Klink algorithm to define the relationships between papers. Rexplore also provides a multiple interlinked view interface which describes the various facets in different views which is also possible using our ConceptCloud browser. The Rexplore tool provides a topic analysis, which graphs how a topic has evolved over time as well as providing an author view which can indicate collaborative patterns. Topic evolution and author collaboration information can also be gathered using our ConceptCloud browser.

8 Conclusions

In this paper we have presented and evaluated an exploratory search tool for a large academic publication collection. Our approach supports exploration of a publication dataset and does not require users to conduct a free-text search of the dataset but rather allows them to dynamically refine the set of publications by selecting tags of interest. While traditional list-based approaches for visualizing publication data require users to examine and manually aggregate a large list of publications, our approach presents users with the aggregated information directly and allows them to make continued refinements until the list of papers is small enough to make manual examination feasible.

We use a combination of concept lattices and interactive tag clouds in order to support exploratory search tasks on publication data. We have also developed a new approach for key-phrase extraction on a large academic publication dataset in order to provide a uniform categorization for the papers in the ACM DL (ACM Digital Library 2016) in order to allow identification of relevant papers according to their topic.

We have also developed a mechanism for supporting relationships between objects using concept lattices, so that we can provide support for browsing the references and citations of academic papers. While the digital library format traditionally shows a linear list of a paper's references and citations, in our approach we present aggregated information from the references and citations. We present aggregated citation and reference data for a single paper (e.g., presenting a summary of topics of the papers that cite a selected paper) and also support aggregation of citation or reference data for a selected property (e.g., authors, topics or conferences) which enables researchers to identify, for example, which authors they reference often or the topics of papers that cite their work. The presentation of aggregated citation and reference data is novel to our approach.

We have conducted a user study to evaluate whether untrained users are able to use our publication browser to answer complex scientometric questions, many of which are not directly supported by traditional indexing services of academic publication data. Our user study shows that users are able to answer complex scientometric questions using our browser with a mean correctness percentage of 73%. We also see that there is a learning effect for new users of the tool which according to White and Roth (2009) correlates well to the exploratory search approach. Additionally, our analysis shows that the prior research experience of the participants does not have a statistically significant impact on their ability to use the tool effectively.

Acknowledgments

This research is funded in part by a STIAS Doctoral Scholarship, CAIR, NRF Grant 93582 and the MIH Media Lab.

References

Abt, H. A., 2007. The future of single-authored papers. Scientometrics 73 (3), 353–358.

Accociation for Computing Machinery, 2015. Acm computing classification system toc. http://www.acm.org/about/class, Online; Accessed 18-Aug-2016.

ACM Digital Library, 2016. Acm digital library. http://dl.acm.org/, Online; Accessed 18-Aug-2016.

Aguillo, I. F., Bar-Ilan, J., Levene, M., Ortega, J. L., 2010. Comparing university rankings. Scientometrics 85 (1), 243–256.

Beck, F., Koch, S., Weiskopf, D., 2016. Visual analysis and dissemination of scientific literature collections with survis. Visualization and Computer Graphics, IEEE Transactions on 22 (1), 180–189.

Brin, S., Page, L., 1998. The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the Seventh International Conference on World Wide Web. WWW '07. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, pp. 107–117.

Carpineto, C., Romano, G., 1996. A lattice conceptual clustering system and its application to browsing retrieval. Machine learning 24 (2), 95–122.

Chen, P., Xie, H., Maslov, S., Redner, S., 2007. Finding scientific gems with Google's PageRank algorithm. Journal of Informetrics 1 (1), 8–15.

Connor, James, 2012. Scholar Updates: Making New Connections. http://googlescholar.blogspot.co.za/2012/08/scholar-updates-making-new-connections.html, Online; Accessed 18-Aug-2016.

Davey, B. A., Priestley, H. A., 2002. Introduction to Lattices and Order (2. ed.). Cambridge University Press.

de Solla Price, D. J., 1965. Networks of Scientific Papers. Science 149 (3683), 510–515.

Dörk, M., Riche, N. H., Ramos, G., Dumais, S., Dec 2012. Pivotpaths: Strolling through faceted information spaces. IEEE Transactions on Visualization and Computer Graphics 18 (12), 2709–2718.

Dunaiski, M., Visser, W., Geldenhuys, J., 2016. Evaluating paper and author ranking algorithms using impact and contribution awards. Journal of Informetrics 10 (2), 392–407.

Dunne, C., Shneiderman, B., Gove, R., Klavans, J., Dorr, B., 2012. Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization. Journal of the American Society for Information Science and Technology 63 (12), 2351–2369.

Eccles, C., 2002. The use of university rankings in the united kingdom. Higher Education in Europe 27 (4), 423–432.

Fischer, B., 2000. Specification-based browsing of software component libraries. Autom. Softw. Eng. 7 (2), 179–200.

Frank, E., Paynter, G. W., Witten, I. H., Gutwin, C., Nevill-Manning, C. G., 1999. Domain-specific keyphrase extraction. In: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence. IJCAI '99. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 668–673.

Ganter, B., 2010. Two basic algorithms in concept analysis. In: International Conference on Formal Concept Analysis. Springer, pp. 312–340.

Ganter, B., Wille, R., 1999. Formal concept analysis - mathematical foundations. Springer.

Garfield, E., 1979. Is Citation Analysis A Legitimate Evaluation Tool? Scientometrics 1 (4), 359–375.

Gollapalli, S. D., Caragea, C., 2014. Extracting keyphrases from research papers using citation networks. In: AAAI. pp. 1629–1635.

Greene, G. J., Fischer, B., 2014. Conceptcloud: A tagcloud browser for software archives. In: Proceedings of the 22Nd ACM SIGSOFT International Symposium on Foundations of Software Engineering. FSE 2014. ACM, New York, NY, USA, pp. 759–762.

Greene, G. J., Fischer, B., 2015. Interactive tag cloud visualization of software version control repositories. In: Software Visualization (VISSOFT), 2015 IEEE 3rd Working Conference on. VISSOFT 2015. IEEE, pp. 56–65.
Greene, G. J., Fischer, B., 2016. Cvexplorer: Identifying candidate developers by mining and exploring their open source contributions. In: Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering. ASE 2016. ACM, New York, NY, USA, pp. 804–809.
Grineva, M., Grinev, M., Lizorkin, D., 2009. Extracting key terms from noisy and multi-theme documents. In: Proceedings of the 18th International Conference on World Wide Web. WWW '09. ACM, New York, NY, USA, pp. 661–670.
Hirsch, J. E., 2005. An index to quantify an individual's scientific research output. Proceedings of the National Academy of Sciences 102 (46), 16569–16572.
Hoey, S. E., 2015. New research features on Mendeley.com! `https://blog.mendeley.com/2015/11/03/new-research-features-on-mendeley-com/`, Online; Accessed 18-Aug-2016.
Huang, C., Tian, Y., Zhou, Z., Ling, C. X., Huang, T., 2006. Keyphrase extraction using semantic networks structure analysis. In: Proceedings of the Sixth International Conference on Data Mining. ICDM '06. IEEE Computer Society, Washington, DC, USA, pp. 275–284.
Hulth, A., 2003. Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. EMNLP '03. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 216–223.
Jiang, X., Hu, Y., Li, H., 2009. A ranking approach to keyphrase extraction. In: Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '09. ACM, New York, NY, USA, pp. 756–757.
Klein, D., Manning, C. D., 2003. Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1. ACL '03. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 423–430.
Li, Y., Bandar, Z. A., McLean, D., 2003. An approach for measuring semantic similarity between words using multiple information sources. IEEE Transactions on Knowledge and Data Engineering 15 (4), 871–882.
Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., Crockett, K., 2006. Sentence similarity based on semantic nets and corpus statistics. IEEE Transactions on Knowledge and Data Engineering 18 (8), 1138–1150.
Lindig, C., 1995. Concept-based component retrieval. In: Working Notes of the IJCAI-95 Workshop: Formal Approaches to the Reuse of Plans, Proofs, and Programs. pp. 21–25.
Lindig, C., 2000. Fast concept analysis. Working with Conceptual Structures-Contributions to ICCS 2000, 152–161.
Liu, P., Wu, Q., Mu, X., Yu, K., Guo, Y., 2015. Detecting the intellectual structure of library and information science based on formal concept analysis. Scientometrics 104 (3), 737–762.
Liu, Z., Li, P., Zheng, Y., Sun, M., 2009. Clustering to find exemplar terms for keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1. EMNLP '09. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 257–266.
Lohmann, S., Ziegler, J., Tetzlaff, L., 2009. Comparison of tag cloud layouts: Task-related performance and visual exploration. In: INTERACT (1). pp. 392–404.
Marchionini, G., 2006. Exploratory search: from finding to understanding. Communications of the ACM 49 (4), 41–46.
Medelyan, O., Frank, E., Witten, I. H., 2009. Human-competitive tagging using automatic keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3. EMNLP '09. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1318–1327.
Medlar, A., Ilves, K., Wang, P., Buntine, W., Glowacka, D., 2016. Pulp: A system for exploratory search of scientific literature. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '16. ACM, New York, NY, USA, pp. 1133–1136.
Mihalcea, R., Tarau, P., 2004. TextRank: Bringing order into texts. Proceedings of EMNLP 4 (4), 404–411.
Miller, G. A., 1995. Wordnet: A lexical database for english. Communications of the ACM 38 (11), 39–41.

Nguyen, T. D., Kan, M.-y., 2007. Keyphrase Extraction in Scientific Publications. In: Proceedings of the 10th International Conference on Asian Digital Libraries: Looking Back 10 Years and Forging New Frontiers. Springer-Verlag, pp. 317–326.

Osborne, F., Motta, E., Mulholland, P., 2013. Exploring scholarly data with rexplore. In: The Semantic Web–ISWC 2013. Springer, pp. 460–477.

Parolo, P. D. B., Pan, R. K., Ghosh, R., Huberman, B. A., Kaski, K., Fortunato, S., 2015. Attention decay in science. Journal of Informetrics 9 (4), 734–745.

Porter, M. F., 1980. An algorithm for suffix stripping. Program 14 (3), 130–137.

Rosvall, M., Bergstrom, C. T., 2010. Mapping change in large networks. PloS one 5 (1), e8694.

Schrammel, J., Leitner, M., Tscheligi, M., 2009. Semantically structured tag clouds: An empirical evaluation of clustered presentation approaches. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '09. ACM, New York, NY, USA, pp. 2037–2040.

Van Dogen, S. M., 2000. Graph clustering by flow simulation. Ph.D. thesis, University of Utrecht.

Wallace, M. L., Larivière, V., Gingras, Y., 2012. A Small World of Citations? The Influence of Collaboration Networks on Citation Practices. PLoS ONE 7 (3), e33339.

Wan, X., Xiao, J., 2008. Single document keyphrase extraction using neighborhood knowledge. In: Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2. AAAI'08. AAAI Press, pp. 855–860.

West, J. D., Bergstrom, T. C., Bergstrom, C. T., 2010. The eigenfactor metricstm: A network approach to assessing scholarly journals. College & Research Libraries 71 (3), 236–244.

West, J. D., Jensen, M. C., Dandrea, R. J., Gordon, G. J., Bergstrom, C. T., 2013. Author-level eigenfactor metrics: Evaluating the influence of authors, institutions, and countries within the social science research network community. Journal of the American Society for Information Science and Technology 64 (4), 787–801.

White, R. W., Roth, R. A., 2009. Exploratory search: Beyond the query-response paradigm. Synthesis Lectures on Information Concepts, Retrieval, and Services 1 (1), 1–98.

Wille, R., 1982. Restructuring lattice theory: an approach based on hierarchies of concepts. In: Ordered sets. Reidel, pp. 445–470.

Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., Nevill-Manning, C. G., 1999. KEA : Practical Automatic Keyphrase Extraction. In: Proceedings of the fourth ACM conference on Digital libraries. ACM, pp. 254–255.

You, W., Fontaine, D., Barthes, J.-P., 2009. Automatic keyphrase extraction with a refined candidate set. In: IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies. Vol. 1. IET, pp. 576–579.

Zhang, J., Yu, Q., Zheng, F., Long, C., Lu, Z., Duan, Z., 2016. Comparing keywords plus of wos and author keywords: A case study of patient adherence research. Journal of the Association for Information Science and Technology 67 (4), 967–972.